

Compressing and decompressing an image.

## BACKGROUND OF THE INVENTION

The invention relates to a method of generating image values of an output image as set forth in the precharacterizing part of Claim 1. The invention furthermore relates to a device and a computer readable medium containing a program implementing the decompression according to that method.

5 Such a method is known from an publication by G.Campbell, T.Defanti and J.Frederiksen published in the ACM SIGGRAPH 86 conference proceedings, Vol. 20, no. 4 (1986) pages 215-223. This known method is directed in particular at compression of texture maps for use in computer graphics, where the method is useful because it can perform decompression at real time speeds and is compatible with computer graphics  
10 architectures for texture mapping.

The known method is concerned with image compression and decompression. For decompression the output image is divided into a matrix of rectangular regions (typically 4x4 image locations). For each region a set of two color values is defined. For each image location a single bit of information is stored in a pixel map. This bit selects  
15 the image value for the image location. If the bit has a first value, say 0, the image value is a first one of the two colors defined for the region to which the image location belongs and if the bit has a second value, say 1, the image value is a second one of these two colors.

For compression the average luminance of the color values from a source image is determined for each region and the image locations in each region are divided into  
20 two groups according to whether their luminance in the source image is above the average for the region or not. Image locations with a luminance value above the average for their region are coded for example as 1 in the pixel map, the other image locations are coded as 0. Instead of luminance values other properties of the color can also be used: see for example a publication by G.Knittel, A.Schilling, A.Kugler and W.Straszer, titled "Hardware  
25 for superior texture performance" published in Computers and Graphics, vol. 20 No.4 pp 475-481, July-Aug 1996.

The pair of color values for a region is determined by forming two averages, one from the color values of each group respectively.

This method preserves the average luminance in each region and some of

09179290-102798  
86/201-0626/150

the color value variations. The method requires only 4 bits per image location to store the compressed image in case of 4x4 regions and 24 bit color values.

The publication shows how this number of bits per image location can be reduced further by constructing a color look-up table (CLUT) from the color values used in the various pairs, followed by coding of the pairs of color values by reference to the (CLUT). If a 256 color CLUT is constructed only 2 bits are needed per image location. However, the construction of the color look-up table is time consuming and makes compression slow, moreover it will not in general be clear according to which criteria such a CLUT should best be constructed.

### SUMMARY OF THE INVENTION

It is an object of the invention to reduce the amount of time needed to compress the image.

The method according to the invention is characterized by the characterizing part of Claim 1. Thus, the original color look-up table provided for the source image is used to represent the colors used for the sets. It has been found that this look-up table also serves well to encode the colors used for different regions in the compressed image. Thus construction of a color look-up table can be avoided.

An embodiment of the method according to the invention is described in claim 2. In this embodiment selection of colors from the look-up table is performed by selecting a representative image location from the set and using its reference to the look-up table. The average of the colors which of the pixels in the set which is used in the prior art will in general not be in the original color look-up table; therefore using the index of a selected color obviates the need to translate the average to the nearest color. Preferably the representative image location is the image location whose luminance is the median luminance among the image locations in a region that belong to the same group. (A median value among a collection of values is defined in statistics as a value whose rank according to size is halfway in the collection). However, acceptable results can also be obtained by taking other image locations, whose rank is not extreme (neither the smallest nor the biggest luminance) or according to a ranking by another property than luminance, e.g. a different weighted average of the color components.

In another embodiment of the method according to the invention the

09179290-102798

output image is a pyramid of images of increasing resolution. By coding all of the levels of this pyramid with their own regions and pixel maps, all referring a common look-up table a high rate of compression can be realized.

The method according to the invention and its embodiments may be implemented using a suitably programmed computer, or using dedicated circuits. The program may be supplied on a computer readable medium such as a disk (hard disk, floppy disk, CD-ROM etc.) or a PROM memory. The program may be provided e.g. in a DLL (Dynamic Link Library) or COM for a PC.

10

### BRIEF DESCRIPTION OF THE DRAWINGS

These and other objects and advantageous aspects of the method according to invention will be described further using the figures in which

figure 1 shows a device for decompressing an image

15

figure 2 shows a flow-chart for compressing an image

### DESCRIPTION OF THE PREFERRED EMBODIMENTS

Figure 1 shows a device for decompressing an image. The device contains a location selector 10 coupled to address inputs of a pixel map memory 12, a first color index memory 13a and a second color index memory 13b respectively. The pixel map memory 12 is coupled to a pixel map multiplexer 14, which has a control input coupled to the location selector 10. The first and second color index memories 13a, b have outputs coupled to a color index multiplexer 16, which has a control input coupled to an output of the pixel map multiplexer 14. The color index multiplexer 16 has an output coupled to an address input of a color look-up table memory 17, which has an output coupled to a display device 18.

30

In operation, an image is conceptually divided for example into a matrix of rectangular regions of 4x4 locations. The pixel map memory 12 has a memory location for each region and that location contains one bit for each location in that region, for example 16 bits in total for a 4x4 region. The first and second color index memory 13a,b each have one memory location for every region and those locations each contain an index pointing at a location in the look-up table memory 17. The index is for example 8 bit long and capable of selecting one of 256 colors from the color look-up memory 17. Each location in the color look-up memory contains a color value comprised of for example three 8-bit

values for red, green and blue intensities respectively.

The location selector 10 selects a location and outputs an indication of the region to which that location belongs to the pixel map memory 12, and the first and second color index memories 13a,b. In response, the pixel map memory outputs the bits for the  
5 locations of the region and the color index memories 13a,b output the indices stored for that region.

The location selector 10 outputs a signal to the pixel map multiplexer 14 to select the bit for the selected location from the bits output from the pixel map memory 12 for the region. The selected bit is supplied to the color index multiplexer, where it selects  
10 one of the two color indices output by the color index memories 13a,b. That selected index is supplied to the color look-up table memory where it addresses a color value that in turn is supplied to the display device where the corresponding color is shown in an image.

As a result the displayed image contains a pattern containing only two colors in each region of 4x4 locations. These two colors may differ from region to region.  
15 The amount of memory needed to store the image is one bit per image location in the pixel map memory 12, 16 bit per region (that is, an average of 1 bit per image location) in the color index memories 13a,b. Furthermore, 256 24-bit locations are needed for the entire image in the color look-up memory 17.

Of course the device shown in figure 1 serves only to illustrate the  
20 decompression process. Different devices which execute the same decompression may be used. For example, the various numbers in the preceding are given by way of example only. For example, the various memories 12, 13a,b, 17 need not be separate memories. Furthermore, the numbers used to discuss the decompression are by way of example only. For example, without deviating from the invention, one may use different sized or shaped  
25 regions, one may use more than two color indices per region, in which case more color index memories 13a,b will be used and more than one bit will be stored per image location in the pixel map memory 12 (e.g. four color index memories 13a,b and two bits per image location) or one may use a different representation for colors in color look-up memory 17.

Figure 2 shows a flow-chart for compressing an image so that it can de  
30 compressed by a device as illustrated in figure 1. In a first step 20 a source image is provided by reference to a color look-up table. That is, for each location in the source image an index is provided that points to a color in the color look-up table. A pixel map with one bit location for each image location is initialized. The source image is divided into a matrix of regions of for example 4x4 image locations. In a second step 22 it is tested whether all

09179290 102798  
854207 0626260

regions have been processed. If this is the case, the flow-chart finishes, otherwise a third step 24 is executed.

In the third step 24, an unprocessed region is selected and the average luminance of the color values of the image locations in this region is determined. This may be done by reading the indices for the image locations in the region, fetching the color values pointed at by these indices in the color look-up table, computing the luminance from these color values (the luminance is a well known linear combination of the RGB components of a color value, or a simple approximation like  $2*R+4*G+B$  which requires only a shift operation to implement the multiplications) and summing the luminance's.

In a fourth step 26, the image locations of the selected region are divided into two groups, a first group containing image locations that have a luminance above the average and a second group containing the remaining image locations from the selected region. The pixel map for the source image is updated by marking the image locations of the selected region according to whether the locations belong to one group or the other.

In a fifth step 28, two color indices are selected for the selected region. A first color index is selected by selecting a representative image location from the first group. This representative image location is an image location whose luminance has a median value among the luminance's of the image locations in the first group. A median value in a collection of values is defined in statistics as a value halfway a ranking of the values according to size. So, for example in the collection of values (144, 510, 15, 831, 200) the values can be ranked by size: (15, 144, 200, 510, 831) and the value halfway is 200. In this case, an image location having luminance value 200 would be selected as representative pixel. (in case of an even number of values any one of the pair of values halfway the ranking may be selected). The index which defines the color value of the representative image location in the source image is used as the index into the color look-up table for all image locations in the first group. The index for the second group is selected similarly. Both indices are recorded for the selected region for use during decompression.

In a sixth step 29, the selected region is marked as processed and the flow-chart returns to the second step 22.

Thus the color values used to represent the image locations in a region after decompression correspond to two color values of image locations in the source image before decompression and these color values are represented by indices into the color look-up table used for the source image. The use of the median luminance as a criterion for selecting the representative image location for choosing the two color values has been proven to give

09179990-102799

good results. However, without deviating from the invention one may also use other rankings, but it is preferable to avoid use of image locations with extreme luminance's (the highest or lowest luminance in the group). This prevents susceptibility to noise and exceptional variations in luminance.

5 Furthermore, other properties than luminance may be used to rank the values, for example only the red color component or other linear combinations of the color components instead of the luminance. The property and rank used may even be different for different regions. For example, if a region contains less than a predetermined range of luminance values, one may use another linear combination of color components to rank the  
10 image locations.

However, if the same criterion is used to select the representative image location in all source images the median of the luminance values has proven to give the best decompression result for most source images.

15 The device of figure 1 and flow-chart of figure 2 are preferably used in the context of texture mapping for computer graphics. Computer graphics imaging of a three dimensional world onto a two-dimensional image provides for a coordinate map which relates coordinates of image locations in a displayed image to coordinates of image locations in a texture image. At an image location in the displayed image a color value from the texture image is shown which originates from an image location to which the image location in the  
20 displayed image maps. Preferably the texture image is compressed as described hereinabove, and the color value for a given displayed image location is obtained by mapping to a texture image location in the texture image and decompressing the texture image for that texture image location as described.

25 Textures for computer graphics are often described as mipmaps, that is as a set of subimages which are images of the same texture, but at different levels of detail (also called different levels of (spatial) resolution). These subimages conceptually form a pyramid, each level in this pyramid corresponds to a subimage and from the top of the pyramid downward the subimages have increasing spatial resolution. When the displayed image is generated, it is computed not only from which texture image location the color  
30 value must be taken, but also from which level or levels in the pyramid information should be used. This is done to prevent aliasing effects.

Preferably all subimages in the pyramid are compressed and decompressed in the way described hereinabove, all using the same color look-up table. To do so, at each level of the pyramid a subimage is divided into regions (usually the number of image

09179290-102798

locations per region is the same for all levels, but the distance between the image locations scales with the spatial resolution of the level, so that the extent of the regions becomes smaller as the spatial resolution becomes higher). For all levels the same color look-up table is used and for each region at each level references to that look-up table are selected. For

5 each level a pixel map is constructed.

Upon decompression, the pixel map memory 10 and the color index memories 13a,b store pixel maps and color indices for regions at all levels. The location selector 10 selects not only the image location but also a level in the pyramid (the image location is rounded to obtain a location available at the selected level). The location selector

10 10 outputs an identification of a region corresponding to that location at the appropriate level and this identification is used to address the pixel map memory 10 and the color index memories 13a,b. This may be repeated for different levels and the decompression results may be combined to obtain a color value for the displayed image. Otherwise decompression does not need to differ from decompression of a single image.

09175290-102798